**Problem :**

Melakukan pencarian nilai minimum dalam suatu fungsi, dimana melakukan pencarian nilai minimum di suatu fungsi dapat menggunakan beberapa cara. Dalam tugas ini pencarian nilai minimum pada suatu fungsi di haruskan menggunakan dan mengimplementasikan algoritma ***Simulated Annealing*** untuk menemukan nilai minimum pada fungsi berikut ini :

Dengan batasan −10 ≤ 𝑥1 ≤ 10 dan −10 ≤ 𝑥2 ≤ 10

**Strategi Penyelesaian Masalah:**

Strategi yang di gunakan yaitu membuat atau mengimplementasikan algoritma ***Simulated Annealing*** ke dalam sebuah program sehingga program tersebut bisa mendapatkan nilai yang optimum dari fungsi yang telah di berikan.

Berikut penjabaran algoritma pada program yang di buat untuk mendapatkan nilai minimum dari fungsi di atas :

1. membuat fuction untuk menghitung nilai fn(x1,x2), dimana nama fuction tersebut di dlaam program adalah fn.
2. Membuat fuction untuk menghitung nilai probabilitas dengan parameter current,newState,Tmax,dimana rumus nilai probabilitas di dapatkan dari **exponen(-DeltaE/T)** dimana **DeltaE** adalah **NewState – current.** nama fuction tersebut di dalam program yaitu proba.
3. Menginisialisasikan variable Tawal dan Tmax dengan nilai 100.
4. Menginisialisasikan variable Tmin dengan nilai 0,0001.
5. Menginisialisasikan variable a dengan nilai 0,999.
6. Merandom nilai x1 dan x2.
7. Mengisi variable current dengan nilai hasil fn(x1,x2).
8. Melakukan perulangan selama Tmax > Tmin.
9. Didalam perulangan poin 8 akan di lakukan perulangan sebanyak 100 kali untuk mengecek dan mendapatkan hasil yang optimum pada Tmax tersebut.
10. Melakukan random nilai x1 dan x2
11. Mengisi variable newState dengan nilai hasil fn(x1,x2).
12. Melakukan pengecekan apakah nilai current > dari nilai newState.
13. Jika nilai NewState < current maka current akan di isi nilai dari newState.
14. Jika nilai Newstate > current maka akan di lakukan pengecekan apakah nilai probabilitasnya > dari hasil random antara 0 smapai 1.
15. Jika probabilitasnya lebih besar maka current akan di isi nilai newState.
16. Setelah perulangan 100 kali selesai maka akan ada perubahan nilai Tmax yaitu Tmax\*a.

**Analisis :**

Dari penjabaran algoritma diatas dapat disimpulkan kalau nilai T adalah parameter yang yang sangat berpengaruh terhadap mendapatkan nilai optimum dikarenakan pada saat nilai newState > current maka nilai T akan berperan penting untuk menentukan apakah nilai newState dapat menjadi current atau tidak, selain itu nilai x1 dan x2 juga berpengaruh untuk mendapatkan nilai optimum karena x1 dan x2 menentukan nilai hasil dari fungsi.

**Hasil Percobaan :**

1. **x1 : 8,0649118895**

**x2 : 9,6556140832**

**T : 0,0004564737**

**Titik optimum : -19,2459108168**

1. **x1 : -8.0619300762**

**x2 : 9.6685402608**

**T : 0.0380913926**

**Titik optimum : -19.2469468920**

1. **x1 : 8.0605566152**

**x2 : -9.6554860868**

**T : 0.0133894287**

**Titik optimum : -19.2464225479**

1. **x1 : -8.0476851948**

**x2 : -9.6539949742**

**T : 0.0195241882**

**Titik optimum : -19.2458379883**

**Screenshot :**

